An Interop Publication

ConneXions®

The Interoperability Report

July 1996 Volume 10, No. 7

ConneXions — The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.

In this issue:

ConneXions is published monthly by Interop Company, a division of SOFTBANK Exposition and Conference Company, 303 Vintage Park Drive, Foster City, California, 94404–1138, USA.
Phone: +1 (415) 578-6900
Fax: +1 (415) 525-0194
E-mail: connexions@interop.com

Subscription hotline: 1-800-575-5717
or +1 610-892-1959

ISSN 0894–5926

From the Editor

As the Internet continues to grow at an exponential rate, tools for measuring and managing its performance are becoming increasingly important. In the telephony world, such tools were developed and deployed in a coordinated fashion from the start. With the Internet the approach has been much more ad hoc. However, work is underway to remedy this situation. Merit Network, Inc. is developing a freeware, turn-key ISP network statistics package. This work, known as *Project NetSCARF,* is described by Bill Norton and Andy Adams in our first article.

Each spring, the Swedish Network Users Society (SNUS) organizes a number of interoperability tests in an event called "Interoperabilitet." This year, the event included the building of an Internet microcosm known as "IT Country." Three of the largest Internet Service Providers (ISPs) in Sweden participated in a challenging routing experiment which included a number of "fake" companies, some of which had connections to more than one ISP. This experiment and the lessons learned are described in an article by Patrik Fältström.

In April we published an article on SNA in our "Back to Basics" series. This month, Eddie Rabinovitch provides a follow-up overview on various methods for SNA migration to and integration with TCP/IP.

The *Java* programming language has had a dramatic impact on the World-Wide Web. Anyone involved in the design and development of Web pages is now more or less expected to learn this new language. Thankfully, a number of books have been published to aid students of Java. Four such books were recently issued by SunSoft and Prentice Hall and are reviewed here by Jon Crowcroft.

NetWorld+Interop 96 Atlanta is only two months away. The program guide for this event has just been published and you can get your copy by calling 1-800-INTEROP. Information about the event is also available from our Web site at `http://www.interop.com`. As always, there will be a number of *Birds of a Feather* (BOF) sessions at the conference. If you have ideas for BOF topics, please send e-mail to `ole@interop.com` and I will provide you with more information. I am also looking for a number of volunteers to be part of our *Conference Assessment Team* (CAT). The CAT program is intended for students who are interested in computer networking. CATs will receive complimentary admission to the conference in return for submitting session evaluations. If you know someone who would be available for this task (September 16–20) please have them get in touch with me via e-mail.

Project NetSCARF

by William B. Norton and Andy Adams, Merit Network Inc.

Introduction

Internet performance measurements are becoming increasingly important. The decommissioning of the NSFNET and the transition to the post-NSFNET Internet model in 1994 brought about an increase in complexity at several levels. [1] As the number of *Internet Service Providers* (ISPs) climbs, as the network technology diversifies, and as network topologies grow in size, it becomes increasingly difficult to quantify the stability and reliability of the Internet.

In April 1996, Merit Network, Inc. was awarded a grant to develop a freeware, turn-key ISP network statistics package. The hope is that the ISP community will adopt and help to evolve this package, thus providing the Internet community with consistent and comparable sets of Internet performance reports. This article provides an overview of the *Scion* software package, developed by the *Network Statistics Collection and Reporting Facility,* or NetSCARF, project. *Scion* uses SNMP to collect network management information from network routers, and employs a standards-based client–server architecture to make that information available on the World-Wide Web.

Overview

During the operation of the NSFNET backbone, Merit evolved a set of network measurements and reports characterizing the growth and reliability of the core U.S. Internet infrastructure. Amongst the myriad network performance measures available, the utility of three was agreed upon by the original OpStats working group and the NSFNET regional network operators. These are:

- Number of packets (and octets) sourced and sinked by a network.
- System downtime indicating the percentage of time network equipment was not operational.
- Interface downtime indicating the percentage of time interfaces were not operational.

In its initial release, the *Scion* package collects the data necessary for calculating these three measurements. Future releases will see an expansion of the collected data and calculated measurements.

Scion components

The *Scion* package consists of five components:

- *scollect:* Collects network data from a set of routers.
- *scook:* Preprocesses, or "cooks" the network data into a more convenient, condensed form.
- *scserver:* Delivers the network data in response to client requests.
- *sclient:* Requests network data from the scserver on behalf of a reporting or graphing application.
- *Real-Time Data (rtdata) tree:* A flat-file database which stores the data collected by *scollect.*

The relationship amongst these components is shown in Figure 1.

Scollect

The statistics collector uses the *Simple Network Management Protocol* version 1 (henceforth referred to as SNMP) to collect the data necessary for calculating the aforementioned performance measures. At fifteen minute intervals, *scollect* queries routers for the values of the following six variables: *sysUpTime, ifLastChange, ifInUcastPkts, ifOutUcastPkts, ifInOctets, ifOutOctets.* In addition, the *ifType* and *ipAdEntIfIndex* objects describing network interfaces are collected once per day.

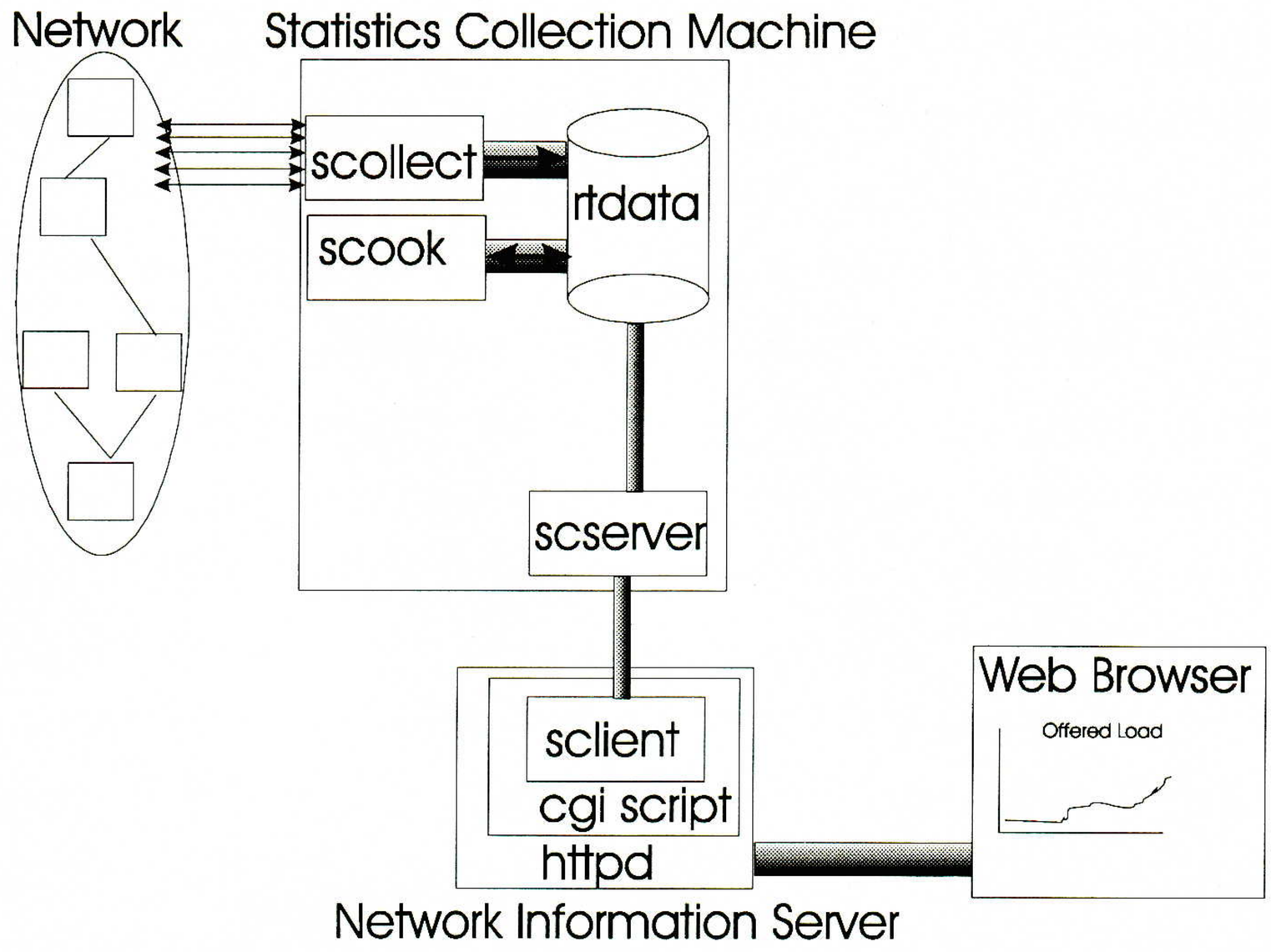


Figure 1: The NetSCARF *Scion* components.

In addition to using SNMP to collect the management information, the *Scion* package can be configured to use the *User-based Security Model for SNMPv2* (described in RFC 1910 and henceforth referred to as SNMPv2u). [6,7,8] The benefits of using SNMPv2u include message authentication and encryption using the familiar user ID and password paradigm. In this model, network operators configure network equipment with a userid, an authentication secret and an encryption secret. The increased security features allow operators to perform *management* operations in addition to monitoring operations. Merit is currently deploying this software in a production capacity for managing the Internet core Route Servers. [2,3,4] Due to export restrictions however, the encryption code and software hooks have been removed from the *Scion* release. Still, there are substantial benefits from using the SNMPv2u with authentication, but detailing the issues and the benefits of SNMPv2u are beyond the scope of this article.

The Fast SNMP Library

Scollect obtains network data via a special SNMP library, developed at Merit. During the management and operation of several medium and large networks, we found empirically that:

> "While continuously polling for fault and performance management information, network management applications typically require instances of all tabular objects of a given type from all devices of a given class."

For example, for performance management, we typically want to collect the interface counters from all interfaces, and to do this for all routers. Furthermore, for both performance and fault management, the more synchronization that occurs during data collection across devices, the easier it is to perform correlation analysis.

These observations led us to develop an optimized SNMP library called the *Fast SNMP Library*. Previously, publicly available SNMP libraries forced an application to wait for a response (or timeout) for a given query before the next query could be issued to a different node.

continued on next page

Project NetSCARF *(continued)*

In contrast, the Fast SNMP API allows the developer to specify a list of nodes, a community string (or user ID/authentication password/ privacy password in the case of SNMPv2u), and a list of management objects to retrieve. The queries are all transmitted back-to-back, without waiting for any replies. The data collection activity is very fast so the results are highly synchronized. The library sets up an empty matrix of hosts and requested objects. The library will query the nodes and fill in the matrix with all data of the requested type.

Two execution threads perform the network queries:

- A transmission loop determines which nodes need to be queried based on last response and the timeout/retry values.
- A receive loop processes the responses and sends follow-up queries as needed to fetch additional objects of the same type.

We found two interesting results from the early deployments of this library.

The Local Router Effect

When transmitting the SNMP queries back-to-back (as opposed to sending a single query to a given router and waiting for that router's reply before transmitting the next request), the local router was often overloaded and dropped packets. This packet loss caused the library to timeout and retransmit queries, resulting in a delay that was especially noticeable for the larger networks we managed.

Natural skewing of responses

While SNMP queries are transmitted back-to-back, nodes receive the queries at slightly different times because of their different topological distances. Responses therefore arrive back at the network management station at slightly different times. This skewing of responses becomes more pronounced as the responses are processed and immediate follow-up queries are sent.

As a result of these two effects, retransmission rates were high for the first iteration of network queries, and substantially lower for the remaining queries. To reduce the local router overload effect, we added code to provide an inter-packet delay. By adjusting this delay, we were able to find an appropriate value (125 microseconds in our case) that minimized the retransmissions while not increasing the polling time substantially. In our experience, this has been highly successful. For a network like the Merit MichNet backbone consisting of about 200 routers, using a timeout value of five seconds and a retry count of five, data collection takes about the expected thirty seconds. It is then the job of the application, in our case *scollect,* to store the data in the `rtdata` tree.

The Real-Time Data Tree

Scollect uses the Fast SNMP Library to query network nodes for specified management information, but where and how should that information be stored? We chose to use the UNIX file system for several reasons:

- *Simplicity:* it was readily available, well documented, and UNIX came equipped with many tools (*grep, find, tar, compress,* etc.) for processing and manipulating the data.
- *Access Control:* the UNIX file system provides access control with the notion of ownership and permission bits.
- *Partitioning:* file systems can be arbitrarily partitioned to prevent the filling of one partition from effecting another.

The architecture begins with a distinction between real-time (continually updated) data and less dynamic (relatively static) data. Less dynamic data need not be archived nightly, whereas real-time data may need to be archived, and perhaps mirrored, on a redundant data collection machine. Thus the `/rtdata` directory is the root directory of the data tree. Since we manage multiple networks, the second level of the tree consists of the names of the managed networks. Hence all data collected from a given network are stored in the subdirectory:

```
/rtdata/<NetworkName>
```

While we are primarily interested in data collected via SNMP, for other networks we collect other types of data. For example, the NSFNET project used *ping* to measure packet delay and used a derivative of the *NNStat* program to collect and classify packet sizes, types, and source/destination pairs. For each of these diverse data collectors, we created a level in the tree. The rule here is that all data collectors store data only in the appropriate subdirectory:

```
/rtdata/<NetworkName>/<DataType>
```

In practice, it is often convenient to have the data collection tasks run as separate user IDs and to have the UNIX file ownership and permission bits govern which process is allowed to write in a given area. The final level of the tree identifies the queried node. Typically this is one of the node's IP addresses. The hierarchy is shown pictorially below in Figure 2.

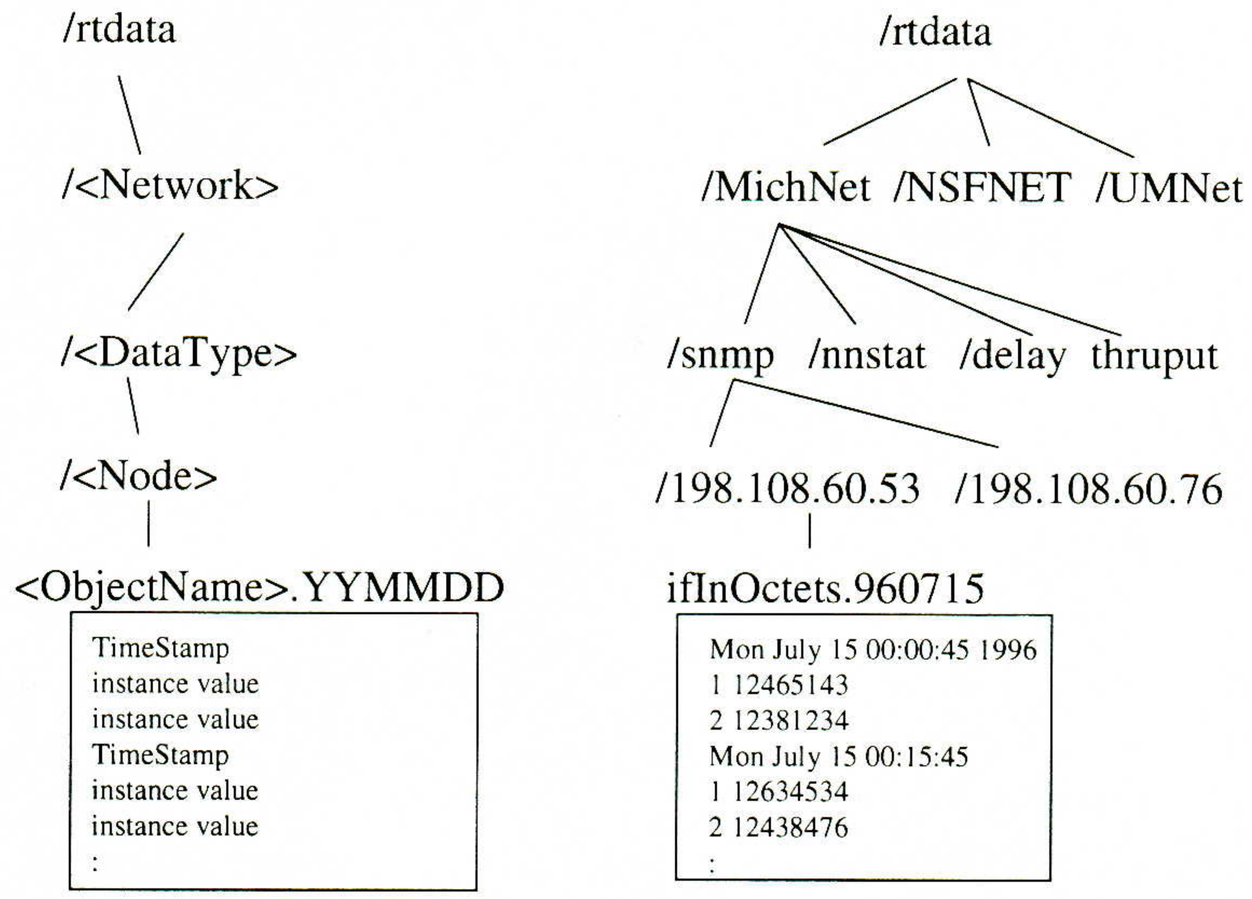


Figure 2: The storage of network statistics data in the Real-Time Data Tree (rtdata)

The names of the files that are used to identify the information they contain. Specifically, the name is of the form `<objectname>.YYMMDD`. In this naming convention, `objectname` is the textual name of the data contained within the file. For data collected using SNMP, this name is the MIB object name. The `YYMMDD` specifies the date that the data were collected.

Each file contains one or more records consisting of one or more lines. The first line of a given record consists of a UNIX timestamp, in Greenwich Mean Time and subsequent lines consist of instance–value pairs of the object type mentioned in the file name.

continued on next page

Project NetSCARF *(continued)*

For example, as shown in Figure 2:

```
/rtdata/MichNet/snmp/198.108.60.53/ifInOctects.960715
```

.....is the name of a file that contains all *ifInOctets* samples for the MichNet node 198.108.60.53 for July 15, 1996. The beginning of that file might look like the one pictured in Figure 2.

There are several data rules governing the contents and the access to files:

- All data contained within the file must be written by one and only one process. This requirement eliminates the need for semaphore file access mechanisms.
- All samples contained within the file must be in time order.
- The timestamp is required for all samples, regardless of whether the node answered the query or not. This allows us to distinguish between a data collector not running and a node simply not responding.

This naming convention provides a name space that is easily searchable using common UNIX tools. For example, one could use the following command to find all files containing data collected in February of 1996.

```
find /rtdata -name "*.9602??" -print
```

This list could then be used for statistics file processing and maintenance activities. One could easily use the above command to archive and remove old data files. In addition, selecting a set of data to be processed is also trivially easy. A find command which locates all **`ifInOctets.96*`** files could be used to process a year's worth of data.

Scook

The *scook* program processes the files of SNMP data written to the **`rtdata`** tree and creates aggregate and summary files. For example, the *sysUpTime* object is a counter of the elapsed hundredths of seconds that a device is up. To create a system downtime report, we care not about each individual *sysUpTime* sample, but rather we care only when the system is reset and how long it remains down.

To this end, *scook* creates an **`sysAggDownTime.YYMMDD`** file containing a single record for each system reset, and a number indicating how long the system was believed to be down. This minimal set of derived data is sufficient for the *sysDownTime* report, and indeed negates the need for the more verbose raw source files. Similarly, the "packets sourced and sinked" report is a summary report that shows the total packets served and sinked by the network. For this report, we don't need the individual fifteen minute samples, but rather the aggregate daily packet counts for all network device interfaces. By creating these compact aggregate files, we greatly speed the data delivery and report processing code.

Scserver and sclient

The *scserver* and *sclient* code proved to be the first public domain implementation of the OpStats (RFC 1856) client–server model. The *scserver* code understands the **`rtdata`** naming scheme and delivers network statistics to the *sclient* for eventual reporting on the World-Wide Web.

Conclusion

We have described the NetSCARF *Scion* software package that allows ISPs to easily and automatically collect and report network performance statistics on the World-Wide Web.

The underlying Fast SNMP Library methodology and a few lessons learned, as well as the Real-Time Data Tree were described. The *Scion* software is available from `http://www.merit.edu/~netscarf/`

References

[1] Susan R. Harris and Elise Gerich, "Retiring the NSFNET Backbone Service: Chronicling the End of an Era," *ConneXions,* Volume 10, No. 4, April 1996.

[2] D. Estrin, J. Postel, and Y. Rekhter, "Routing Arbiter Architecture," *ConneXions,* Volume 8, No. 8, August 1994.

[3] Manning, B., "The Routing Arbiter in the post-NSFNET Service World," *ConneXions,* Volume 9, No. 9, September 1995.

[4] Yu, J., "The RA Route Server Service Overview," *ConneXions,* Volume 9, No. 8, August 1995.

[5] Adams, A., "The MERIT Policy-Routing Configuration System," *ConneXions,* Volume 7, No. 2, February 1993.

[6] Glenn Waters, "The User-based Security Model for SNMPv2," *ConneXions,* Volume 10, No. 5, May 1996.

[7] Rose, M., et. at., "The USEC Resource Page," Available from: `http://www.simple-times.org/pub/simple-times/usec/`

[8] Waters, G., Editor, "User-based Security Model for SNMPv2," RFC 1910, February 1996.

[9] Rose, M. T., "Network Management: Status and Challenges," *ConneXions,* Volume 7, No. 6, June 1993.

[10] Rose, M. T., *The Simple Book: An Introduction to Networking Management,* Revised Second Edition, Prentice-Hall, ISBN 0-13-451659-1, 1996.

[11] *ConneXions,* Two *Special Issues* on Network Management and Network Security, Volume 3, No. 3, March 1989 and Volume 4, No. 8, August 1990.

[12] Stallings, W., "Cryptographic Algorithms," Part I: Conventional Cryptography, *ConneXions,* Volume 8, No. 9, September 1994. Part II: Public-Key Encryption and Secure Hash Functions, *ConneXions,* Volume 8, No. 10, October 1994.

WILLIAM B. NORTON is the NetSCARF Project Director and a Senior Researcher at Merit Network, Inc. Since 1995, he has been the Chair of the North American Network Operators Group (NANOG), a public forum for discussing operational issues of the U.S. Internet. Over the past eight years, Mr. Norton provided operations center support, and led a team of engineers in building the shared Network Operations Center that manages the CICNet, MichNet, and UMNet network infrastructures. Mr. Norton designed and led the implementation team for the Network Management Architecture for the Routing Arbiter. These days, much of his attention is spent in areas of network management and network performance statistics analysis. Mr. Norton has been very active in the industry, giving talks at NetWorld+Interop, INET, NANOG, and various IETF working groups. E-mail: `wbn@umich.edu`

ANDREW ADAMS is the Lead Software Architect for Project NetSCARF. He received a B.S.E. in Computer Engineering from the University of Michigan in Ann Arbor in 1991 and is currently working on a M.S.E. in Computer Science. From 1988 to the present he has worked for Merit Network, Inc., housed at the University of Michigan. While at Merit he has been a principle architect and an implementor for several large software projects in **C** and **C++**, most notably the NSFNET Policy Routing Database System. When Mr. Adams is not working on computer networking projects, he does research on infectious disease epidemics by way of computer simulations and mathematical models. E-mail: `ala@merit.edu`

Interoperabilitet '96

by Patrick Fältström, Tele2

Introduction

Interoperabilitet '96 was an event run by the *Swedish Network Users Society* (SNUS) [1]. Each year since 1991, some kind of interoperability tests have been hosted by SNUS. The first two years were targeted to IP-level conformance on the link level, testing things like ISDN and SNA tunneled over IP. Routing protocols were also tested. Later, interoperability tests of Internet mail (RFC 822 and MIME), Firewalls and other link protocols such as ATM have been introduced.

1996 was a year when a new test was introduced. Apart from tests of different products in the classes "Internet Mail with MIME," "Firewalls" and "ATM," a more general test focusing on how to build an Internet, how to connect a company to Internet, and how to build an Intranet was established.

IT Country

The main task was to invite the Internet Service Providers (ISPs) operating in Sweden, and ask them to first of all extend their backbone to the fair itself. On the floor, several "fake" companies were invited, and each one of them had a connection to the Internet via one (or two) of the ISPs. The intention was to test not only the ISPs but also consultants that have knowledge about exactly this problem—how to connect a company to the Internet and publish services on the Internet. The resulting setup was called the "IT Country."

The "fake" companies were divided into three classes:

- *Class 1:* With only a simple connection to the Internet, possible via a modem, these companies do not publish information, only run protocols like client side HTTP and POP to fetch information from the net. At the fair, these companies had a fixed connection in most cases, because we think the knowledge about running PPP or SLIP over a modem is so well known, and the problems one can get due to hardware problems was not something we wanted to introduce. (Last year we had tests of modem pools, terminal servers, PPP and SLIP, and we saw hardware problems with an analog telephone switch at the fair itself. Making the hardware run was very, very hard, but in real life most of those problems are already fixed by the local ISP when you get your telephone line.)

- *Class 2:* These companies had a fixed line to the Internet. This is probably the most common way of connecting a company to the Internet. The company contacts one ISP, gets a domain name, IP address (one "/24" network, formerly called "Class C"), and agrees on if the company itself or the ISP is going to run DNS and other services. Sometimes, the company buys other services from the ISP, such as encrypted lines and gateways between RFC 822 and X.400.

- *Class 3:* Only three companies were of this class, and of the three, only two participated and reached the goals. The class three companies were connected to two different ISPs at the same time, getting services from both (but maybe not at exactly the same time). The idea was to check what kind of redundancy you get if you get two connections from two different ISPs compared to having two connections from the same ISP. This is the most interesting setup of the three classes.

The participating ISPs were Telia [2], Global-One [3] and Tele2 [4]. They connected each to about 6 "fake" companies. Telia and Tele2 were also responsible for one "fake" company each.

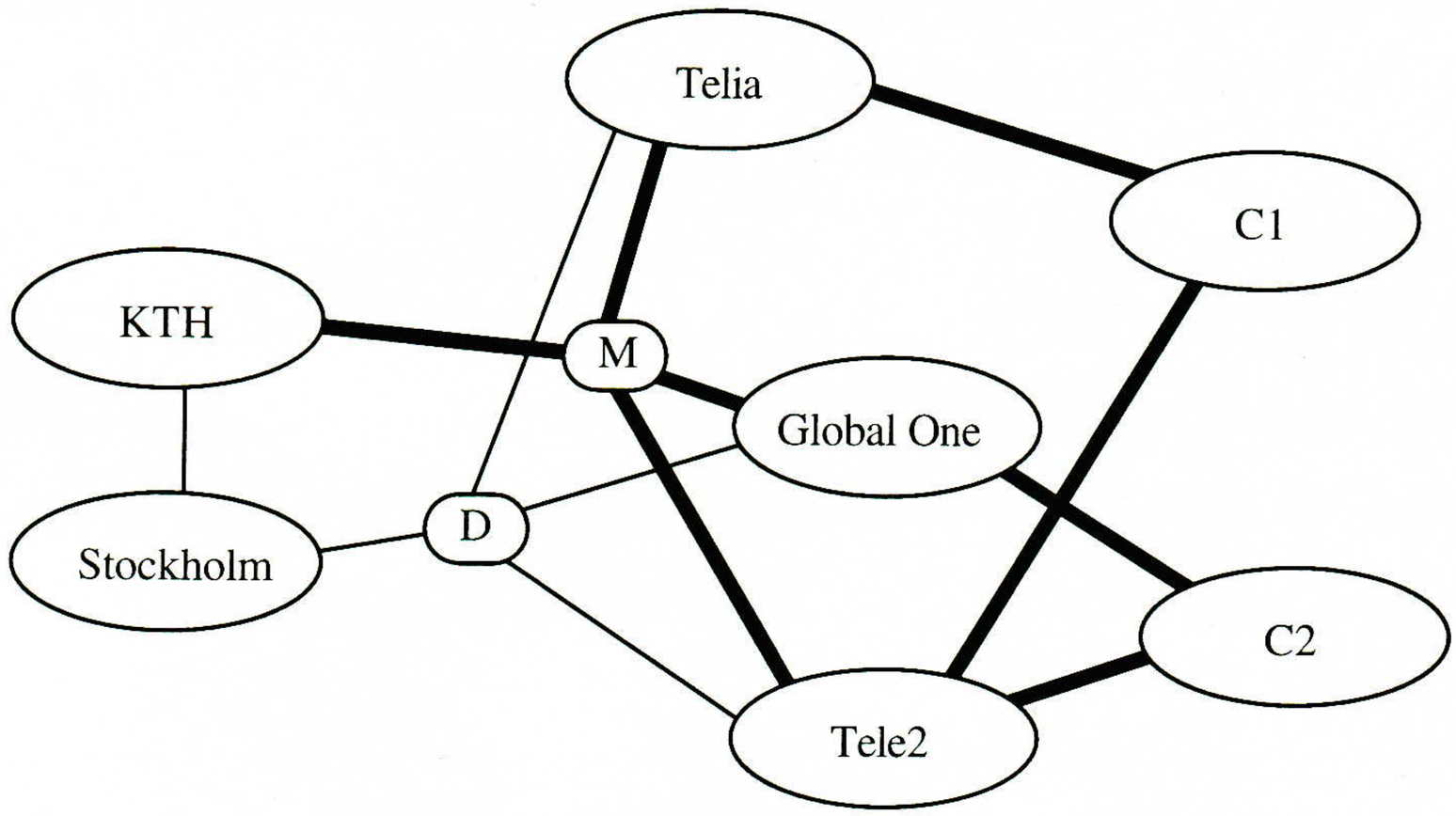
Live backbones

All three ISPs extended their backbone out to the fair, connecting with two full ISDN lines (i.e., a total of 4 times 64kbps per ISP were available). All three of them also extended their ASs out to the fair, so full routing was available. This last step was not intended from the beginning due to the problems the ISPs would get whenever they happened to get a full update of the routing tables over the "slow" ISDN link as a result of some routing flaps. The networks that were available were chosen to really be live, and any mistake could really damage the production in the ISPs' worldwide backbones. We think the ISPs were quite brave to attempt this setup.

The Network Operations Center at the fair (the NOC) was running an exchange point (IX) where the ISPs would connect to exchange traffic and routing information. The NOC also had two ISDN lines to the world via the Royal Institute of Technology's {*Kungliga Tekniska Högskolan* [KTH]} backbone in Stockholm) but one of them was used as a normal telephone, because the need for a phone was higher than the need for more bandwidth beyond 2*64kbps. The fact that we had one IX (called MIX because the Swedish word for "fair" is "mässa" so the acronym stands for "Mässa-IX") at the fair made the routing more complicated as the ISPs already are connected to at least one more IX, the D-GIX at the KTHNOC [5] in Stockholm.

The routing test involved getting routing to work as efficiently as possible after creating a new global exchange point between the ISPs, and at the same time adding some customers to more than one ISP. All of this in one week while tests of higher level services (like RFC 822/MIME mail and HTTP) were done on top of the network.

Locally at the fair, the three ISPs and the NOC had somewhat different hardware. Telia choose to use "digital-x-line" connections to their companies while Tele2 used modems, each connecting one central router with a router at the customer's site. The customer would get one twisted pair connection at the other side of that router. Global-One was using a central router with direct twisted pair connections to the "fake" companies. The NOC had one router connecting (with twisted pair) both the NOC backbone and the MIX, to which the other ISPs were connected (see below).



This picture shows the connections between the active ASs. The thick lines represents what was built at this fair, and the thin ones what is used normally. C1 and C2 represents the two fake companies that were built multihomed.

continued on next page

Interoperabilitet '96 *(continued)*

Multihomed companies

The network management and the services operated by the NOC was performed mainly on two Sun workstations. One running NetBSD [6], the other one Plan9 [7]. Network monitoring was done via SNMP on a Macintosh using the InterMapper [8] software and SNMP Watcher [9]. Normal tools like *traceroute* and *ping* were of course also used, from both the computers mentioned above, and other x86 based PCs running BSD/OS [10] and NetBSD. The CAP package [11] was used on the Sun running NetBSD to make it possible to print on an old Apple LaserWriter that only had support for EtherTalk. We were running DNS, HTTP, FTP, Whois++ [12] and SMTP daemons to facilitate the tests at the fair.

Back to the interesting routing setup. Each ISP had their "live" AS extended to the fair, which means that on the MIX, the ISPs were supposed to exchange routing information between the ASs the same way they do at the D-GIX at KTH. The first shot, to make the IP work, was to run EIGRP among all the routers at the fair, but that was a boring exercise. We had different ISPs, so they should run as separate ones. On day three we started to use BGP4 [20] instead, with live AS numbers. On the MIX, four ASs did meet, and the two companies that were connected to two ISPs had their own AS number which forced the ISP and the "class three company" to also exchange routing information (more about that later).

Best path in BGP4?

According to the theory, BGP4 [13] only selects the best path, and is not doing any load balancing, and this was of course also what happened at the fair. Another theoretical thing we could show was that a company which is connected to two ISPs, one primary and one secondary, should get their IP address out of the CIDR block of the secondary ISP. This because the ISP from which the network is not chosen will make an explicit announcement of the block, and BGP4 will subsequently choose this path before the one where the announcement is aggregated into the announcement from the ISP itself. Longest match is what rules, and we were able to show that it actually works that way, something not many people have seen live.

The filters that had to be set up between the ISPs were not simple. This because the ISPs were connected not only at one IX, but two (at least) and as well at the multihomed companies. Asymmetric traffic was the result before the routing filters were set up correctly—another interesting exercise for people not working with this daily.

One of the multihomed companies did complicate the situation a little bit more. They had one router connected to each of their two ISPs (Telia and Tele2), and each router had its own firewall behind it. Between the two firewalls was an internal network which was set up according to all the standard rules regarding DNS, proxies etc. The interesting task here was to exchange routing information between the two routers through the two firewalls. This was achieved, but due to some bugs in the routing software at one of the ISPs, the company was never completely connected to two ISPs simultaneously. We suspect that with one more day of testing (or maybe one more hour), even that problem would have been solved.

Discussion

What are the conclusions then? First of all, one does not get what one wants when multihoming to more than one ISP. The resulting redundancy as compared to connecting two lines to the *same* ISP is desirable in some sense, but due to the lack of load balancing in BGP4, the backup link is only a backup link—which probably costs too much to have in the long run.

Having two lines to the same ISP makes it possible to participate in the same AS as the ISP itself, and thus use the two lines more efficiently. If a customer has his own AS number (as probably will be the case when doing this kind of multihoming) he has to be prepared to really participate in the manual setup of the routing filters between the AS clouds. The customer basically becomes a small ISP himself.

The ISPs participating in the fair also pointed out that this kind of multihoming is not something they have in their price list, and not something they would recommend to any customer. Maybe very large organizations, such as a multinational company, could be connected to more than one ISP, but the question is does this means that every computer on the customer's network can reach the outer world via any of the two links? That is probably not what they really want because the setup is very, very complicated. It is much easier—especially internally for the customer—to have one default route for each point on their network.

The responsible persons for IT-country were Lars-Johan Liman (`liman@sunet.se`) and Patrik Fältström (`paf@swip.net`).

RFC 822/MIME

The MIME tests were done mostly before the actual fair. Participants completed a form (see `http://www.cafax.se/MIME-tests.html`) and also used a mailback server. The mailback server resides at Bunyip Information Systems in Montreal. Send e-mail to the address `mimeback@bunyip.com` with the single word "HELP" in the Subject: line, and you will see what tests that can be handled by this server.

Participating companies and products were:

DES Communications, Borderware Firewall Server, 3.1.1
GEIS Sweden, Business Network, 2.3
ICL, EMBLA, 1.2
Matti Aarnio–FUNET, Zmailer, 2.99.27
Microsoft AB, Microsoft Exchange Server, 4.0 (build 4.0.837.0)
NetGain, NetGain Mimetic, 2.0
Sun Microsystems AB, Pronto E-mail, 2.0.1
Sun Microsystems AB, Solstice Internet Mail client, 0.9
TeamWARE Group, TeamWARE Internet Mail, V5
Tele2, ADMD InterX X.400-to-Internet Gateway

The software packages that preliminary passed the tests were EMBLA, Zmailer, Microsoft Exchange Server, NetGain Mimetic, Pronto E-Mail and InterX X.400-to-Internet Gateway.

Responsible persons for the RFC 822/MIME tests were Patrik Fältström (`paf@swip.net`) and Lars-Johan Liman (`liman@sunet.se`)

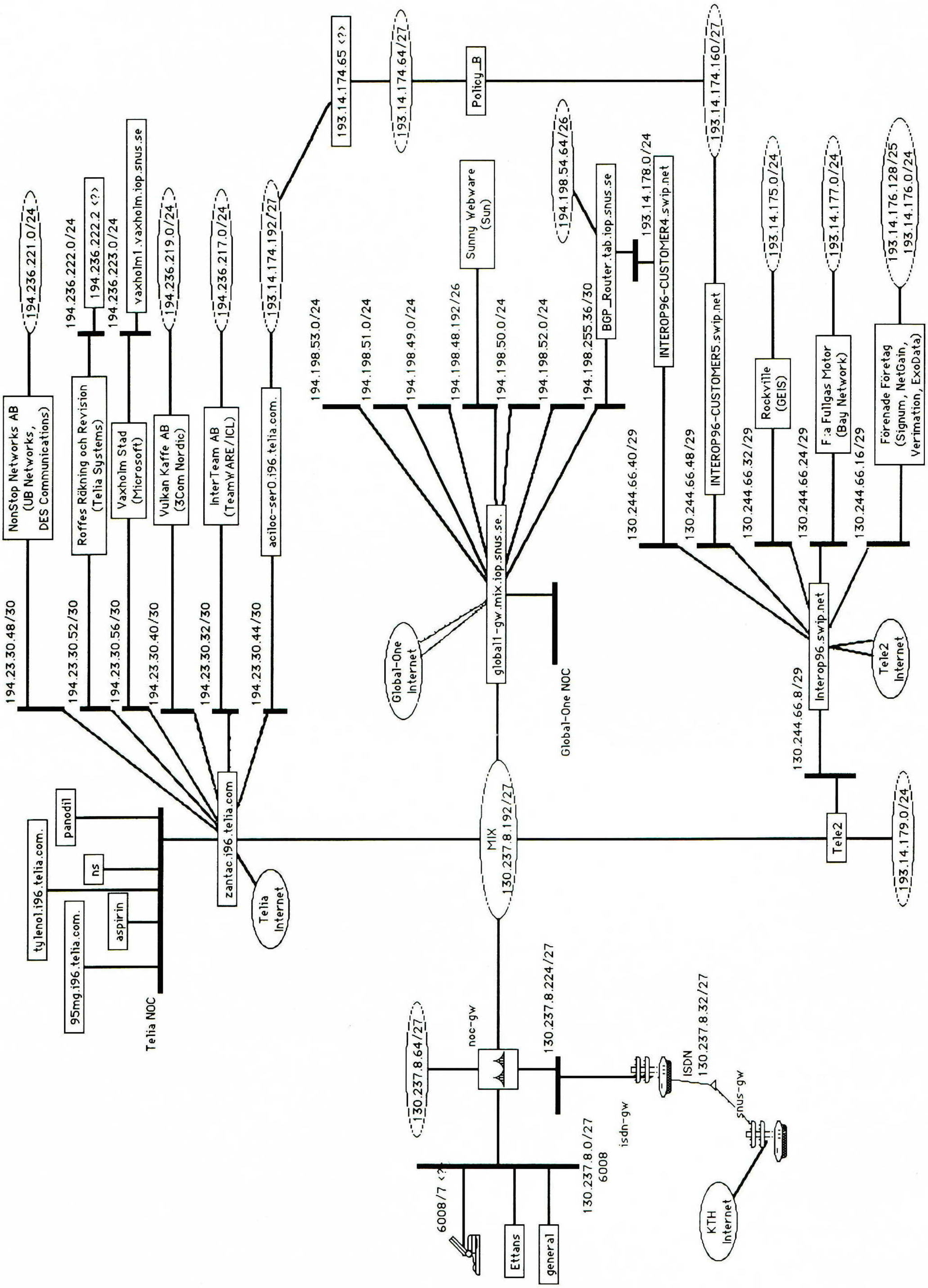
Another test involved LAN Emulation over ATM. The objective with these tests was to give the data communication specialists the ability to test ATM LAN Emulation products from different vendors against their own products. The goal was to get knowledge about different implementations of ATM LAN Emulation.

ATM LAN Emulation

The LAN Emulation tested the functions LECS, LES, LEC and BUS between different vendors of ATM switches. The tests also included ATMswitch to ATMswitch between different vendors. The conclusion is that almost every ATM LEC interoperates with the tested services provided in the ATM network. The problems encountered in these tests depended on configuration rather than a mismatch in the ATM LANE phase 1 standard. ATM Forum phase 1 works well according to our tests, especially the ATM LAN Emulation clients.

continued on next page

Interoperabilitet '96 *(continued)*



The actual network at the fair as it was just before closing time.

Participating companies and products in the ATM test were:

AU-System, Forerunner ES-3810, Forerunner ASX-200WG, Forerunner PC card
PCA-200E
Bay Networks, 5000AH, Ethercell 10328-F, Wellfleet BLN VNR
Cisco Systems, Catalyst 5000, Cisco 7010, Cisco LS100, Cisco LS 1010
Telia Systems, First Virtual FSW1000, First Virtual FVC ISA+
Sun Microsystems, Sun ATM 1.0 SBus-Card
3Com, Cellplex 7000, Linkswitch 2700
UB Networks, GeoSwitch, Interphase 4615, ATM Sbus Adapter

The responsible person for the ATM tests was Niklas Gerdin, Frontec Network Services AB, (`Niklas.Gerdin@sth.frontec.se`).

Firewalls

The final test involved firewalls. At the time of writing of this article, no results from the tests were available. I have to point to the final report from the Interoperabilitet '96 event for these results.

Responsible person for the firewall tests was Staffan Hagnell, Network Management, (`shl@netman.se`).

More information

For more information about the Interoperabilitet events and the Swedish Network Society, send e-mail to `info@snus.se` or have a look at the Web pages at `http://www.snus.se/SNUS/.`

We would like to thank Softbank Expos for sponsoring this event.

In conclusion, we had fun and we look forward to another Interoperabilitet next year!

References

[1] SNUS—The Swedish Network Users Society. ("Snus" is also Swedish for "snuff" or chewing tobacco, and small cans of it was distributed at the fair as a hack). `http://www.snus.se/SNUS/`

[2] Telia—One of three ISPs participating at Interoperabilitet '96. `http://www.telia.se/`

[3] Global-One—One of three ISPs participating at Interoperabilitet '96. `http://www.global-one.se/`

[4] Tele2—One of three ISPs participating at Interoperabilitet '96. `http://www.tele2.se/`

[5] KTHNOC—An organization running one of the most well connected Internet Exchange points in the world. `http://www.sunet.se/kthnoc/`

[6] The NetBSD Project is the collective volunteer effort of a large group of people, to produce a freely available and redistributable UNIX-like operating system, NetBSD. NetBSD is based on a variety of free software, including 4.4BSD Lite from the University of California, Berkeley. `http://www.netbsd.org/`

[7] Plan 9 is a new computer operating system and associated utilities. It has been built over the past several years by the Computing Sciences Research Center of Bell Laboratories, the same group that developed UNIX, **C**, and **C++**. `http://plan9.att.com/plan9/`

continued on next page

Interoperabilitet '96 *(continued)*

[8] InterMapper is a Macintosh program that monitors IP and AppleTalk internetworks. It reports configuration changes, interface errors, and reachability problems detected using the Simple Network Management Protocol (SNMP).
`http://www.dartmouth.edu/pages/softdev/intermapper.html`

[9] SNMP Watcher is a tool for retrieving information using the Simple Network Management Protocol (SNMP) on a Macintosh.
`http://www.dartmouth.edu/pages/softdev/snmpwatcher.html`

[10] BSD/OS is a full-function, POSIX-compatible, UNIX-like system for the 386, 486, Pentium architectures. It is based on the BSD software from the University of California at Berkeley, a number of other sources, and components engineered by BSDI.
`http://www.bsdi.com/`

[11] The Columbia AppleTalk Package (CAP) implements the AppleTalk protocol stack on a variety of UNIX machines.
`http://www.cs.mu.oz.au/appletalk/cap.page`

[12] C. Weider and P. Fältström, "The WHOIS++ Directory Service," *ConneXions,* Volume 8, No. 12, December 1994. See also `http://www.bunyip.com/products/digger`

[13] For more information about how to do Inter-Domain routing, see *ConneXions,* Volume 5, No. 1, January 1991—*Special Issue:* Inter-domain Routing.

[14] Laubach, Mark, "ATM for your internet—But When?" *ConneXions,* Volume 7, No. 9, September 1993.

[15] Atkinson, Ran, "Towards Real ATM Interoperability," *ConneXions,* Volume 7, No. 8, August 1993.

[16] Laubach, M., "IP Over ATM and the Construction of High-Speed Subnet Backbones," *ConneXions,* Volume 8, No. 7, July 1994.

[17] Heinänen, Juha, "Multiprotocol Encapsulation over ATM," RFC 1483, March 1993.

[18] Stockman, B., "Global Connectivity: The Global Internet Exchange (GIX)," *ConneXions,* Volume 7, No. 11, November 1993.

[19] Fuller, V., Li, T., Yu, J., Varadhan, K., "Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy," RFC 1519, September 1993.

[20] Rekhter, Y., and Li, T. (Editors), "A Border Gateway Protocol 4 (BGP-4)," RFC 1654, July 1994.

[21] Radia Perlman, *Interconnections: Bridges and Routers,* Addison-Wesley, Reading, Massachusetts, 1992.

PATRIK FÄLTSTRÖM is a senior researcher at Tele2 in Stockholm, Sweden. He has a MSc in Mathematics from the University of Stockholm, and has been working with standardization on the Internet for some 10 years. He is author and co-author of RFCs regarding MIME content types and the directory service Whois++, and is chair of the FIND working group in the IETF. He is also chair of AG12 in Sweden which is a group responsible for rules regarding electronic mail addresses in Sweden. He is working with distributed indexing services, such as Whois++ and the Common Indexing Protocol, at Tele2, together with Bunyip Information Systems in Montreal and KTH in Stockholm. He can be reached at the e-mail address `paf@swip.net`.

Making Migration to TCP/IP Seamless

by Eddie Rabinovitch, Unisys Corporation

Introduction

The jury is still out on the question which network architecture will survive into the next millennium? One answer is definite: it's not going to be the full-blown OSI model as some believed few years ago.

TCP/IP and SNA were both designed from diametrically opposite standpoints: SNA used to be a centrally managed architecture with predefined static routing aimed at high bandwidth utilization and optimal response time for the users, whereas TCP/IP is a completely decentralized network with dynamic routing aimed at easier connectivity for the end users.

In many cases DSS (*Decision Support Systems*) client–server applications, which by nature are not transaction oriented, usually would not require synchronization points, two-phase commits, or concurrent updates of distributed databases. Since most of the SNA/ APPN advantages are not applicable in such environments many have chosen TCP/IP as the underlying network protocol for better positioning of upcoming client–server applications. However, as for any good rule—there obviously are some exceptions to it: we can probably identify some cases where SNA/APPN (with *High Performance Routing* [HPR] when it becomes available) would make more sense than TCP/IP. For example, if:

- The client application runs on a customer's PC that requires an SNA/ APPN connection: since "the customer is always right," we probably would not recommend to argue with him or her.
- A distributed client–server transaction requires concurrent updates of distributed databases. The server application resides on an SNA mainframe or an AS/400.
- There are existing off-the-shelf APPC interface tools, where implementation of TCP/IP would require purchasing of additional software and/or hardware gateways. (I know, it sounds unlikely, since what product today would not have C-socket interfaces, but theoretically it is possible).
- New client–server applications will have to share the backbone network with existing legacy SNA applications for a *considerable* amount of time, thus securing comparable service levels for interactive legacy applications.
- There is a specific requirement for a higher priority (Class Of Service) for a selected group of applications (e.g., interactive) that share a *highly utilized bandwidth* with applications less sensitive to response time (e.g., file transfer).

Only a few years ago many analysts and industry watchers claimed a "premature death" for SNA. However, based on the recent statistics, these estimates seem to be a "bit" over-exaggerated. Although the number of TCP/IP installations is growing at an exponential rate (basically, because almost any PC based communications package includes the IP stack), not too many of 50,000+ SNA installations migrated their mission critical applications to IP. On the contrary, IDC's survey shows that the number of SNA gateway installations in 1994 grew more than 16% from 279,000 to 324,000.

So, most likely, for the next several years the network infrastructure for most organizations will include a combination of SNA/APPN and TCP/IP.

continued on next page

Making Migration to TCP/IP Seamless *(continued)*

Since not too many companies have the luxury of turning off their existing network infrastructure and waiting for the emerging new single networking architecture—the real question for network managers today is: what is the most effective way of supporting new applications over the existing infrastructure? Fortunately for the users, some vendors have addressed this question and provide us with real answers here and now! Let's analyze the characteristics of TCP/IP and SNA methods and see the solutions for seamless integration and migration to TCP/IP.

TCP/IP routing

Following a successful deployment of new applications on the LANs, there was a need to connect them over WANs. Shortly after, the developers and users had realized that, unfortunately, the bandwidth of the WAN was not sufficient for LAN based protocols. Most of the LAN protocols (AppleTalk, NetBIOS, IPX) are too "chatty" and may cause so called "broadcast storms." Therefore, the next natural step was to utilize the least expensive and most available protocol for WAN communications. However, there was no magic with TCP/IP either: the overhead was significantly higher than what it was with SNA. And while this was not a real problem on LANs, it became a major headache on the WANs. An additional problem involved TCP/IP's use of congestion detection as opposed to congestion avoidance used in SNA. Therefore, TCP/IP networks are more susceptible to runaway overloads at high line utilization. Further problems are introduced by the "ancient," but still the most popular IP Routing Information Protocol (RIP). As a distance-vector protocol, RIP is inflexible, it introduces additional overhead and can create "loops" and "black-holes" in complex IP networks.

Routing with SNA

The traditional (pre-APPN) SNA, commonly associated with 3270 applications, is a connection-oriented protocol with routing only at the fourth—Transport—Layer of the OSI model (equivalent to the Transmission Control Layer in SNA). However, this layer lacks dynamic rerouting capabilities, and high intelligence is required in all routing nodes. The APPN today is usually associated with application-to-application (peer-to-peer) networking. It employs a connection-oriented *Intermediate Session Routing* (ISR) link state protocol. Link state routers (network nodes) are characterized by a topology database, containing information about states of all network links, that are included in each network node. Each node is responsible for computing the best routes to any other node in the network. APPN Network Nodes are responsible for maintaining the topology database, the integrity of which is preserved by using sessions between adjacent Network Nodes. This is different from distance-vector based protocols (i.e., RIP), where such information is periodically broadcast (every 30 seconds with RIP), regardless of whether there were any changes or not. The more advanced TCP/IP link-state based routing protocols (such as OSPF) also broadcast their routing information when a link state is changed. However, they also periodically broadcast this information (every 30 minutes) regardless of changes in the link states. When a session is established between two nodes in APPN network it will live as long as the preferred route, selected during the session setup, is available. The major drawback of this method is that the preferred route, during a long lived session setup, will not necessarily remain the optimal route for the life of the session. There is also no rerouting around failures (i.e., sessions have to be restarted even due to an intermediary node failure), which may potentially disrupt a client–server application.

To address the major concerns of SNA/APPN, IBM came out with an enhancement called *High Performance Routing* (HPR). HPR is a connectionless protocol, allowing dropping packets during network congestion and non-disruptive rerouting around failures. The *Rapid Transport Protocol* (RTP) is used at the end-points of an APPN HPR session to guarantee delivery and re-sequencing of packets. An *Adaptive Rate Based* (ARB) flow control mechanism is used at the end points of an HPR session to prevent network congestion. HPR can be mixed with ISR nodes within an SNA/APPN network.

Trends in LAN Connectivity

Since the number of TCP/IP based client–server applications and LAN based applications is constantly growing, in most cases one can find a mix of SNA and non-SNA (i.e., TCP/IP, IPX/SPX, NetBIOS, Vines, AppleTalk, etc.) networks next to each other. Such network strategy is probably safe but *too costly!* Rather than maintaining parallel networks for different protocols, networks can now be combined without impacting application support.

Many vendors realized the importance of consolidation of the network infrastructure and the potential savings it can bring to the users and developed products, allowing utilization of non-SNA networks for SNA traffic. The most popular approaches for multi-protocol network consolidation, include SNA transport over IP with *Data Link Switching* (DLSw), Frame-Relay with RFC 1490, and LAN transport over SNA/APPN backbones.

DLSw

Multi-protocol routers with DLSw are probably some of the more important examples of the SNA over IP approach, which was originally introduced by IBM, and in March of 1993 adopted by the Internet Engineering Task Force (IETF) as RFC 1434. This RFC for DLSw was enhanced and superseded by RFC 1795 in April 1995. Because of the high timing sensitivity of SNA sessions, DLSw is the only dependable technique of transporting SNA over IP networks. Actually, encapsulated SNA data into IP frames is not new to the router vendors. However, similar pre-DLSw attempts ran into timeout problems. Therefore, IBM created DLSw to address this shortcoming, which reduces the likelihood of timeouts, by providing a local polling and logical local termination of the data link, often named "spoofing." By keeping the acknowledgment local, routers also reduce the amount of traffic traversing the wide area link. Additional advantages of DLSw include the ability to route around link failures, support for an extended source route bridging hop count, and NetBIOS name caching, that highly reduces NetBIOS broadcasts.

But DLSw is certainly not a universal solution for consolidation of SNA and multi-protocol networks. This is mainly because it does not resolve the shortcomings of TCP/IP networks compared to SNA. For example, when SNA data is encapsulated and sent over IP, there are no provisions for defining SNA Class of Services (CoS) necessary to secure prioritization of interactive sessions versus file transfer or printing. There is also a significant overhead of IP framing that adds more than 50 bytes to each encapsulated frame.

RFC 1490 for Frame-Relay

The increasing popularity of Frame-Relay networks introduced a new alternative for networks consolidation—RFC 1490. It allows transport of multiple protocols over the same access line and even the same virtual circuit—*Data Link Connection Identifier* (DLCI). The overhead associated with RFC 1490, for transporting SNA traffic, is about one fifth of the DLSw overhead.

continued on next page

Making Migration to TCP/IP Seamless *(continued)*

Some vendors, of Frame-Relay access devices, allow separate queues for different protocols and a protocol-level prioritization scheme. However, within the Frame-Relay "cloud" there are no provisions for neither protocol prioritization nor SNA CoS.

Although, RFC 1490 supports protocol mixing for a single DLCI, in most cases it may be advantageous to keep the SNA traffic on a separate DLCI, with a higher committed information rate, than the less critical traffic. Planning for Frame-Relay access lines consolidation is a critical issue that has to be very carefully addressed. One must take all precautions to isolate mission-critical traffic from less critical applications.

Specification of the Frame-Relay flow control standard is quite vague, which obviously caused inconsistencies with implementation of indications for *Forward Explicit Congestion Notification* (FECN), *Backward Explicit Congestion Notification* (BECN), and *Discard Eligibility* (DE) bits among different vendors. FECN or BECN is set in the frame address to notify the *Frame-Relay data Terminal Equipment* (FRTE) users of a congestion condition on this DLCI's queue. At this point, however, frames will not be discarded until severe congestion is reached—setting the DE bit on. Before severe congestion occurs, the FRTE devices have a chance to help relieve congestion by reacting to the FECN/BECN bits with the opening and closing of network window sizes. The FECN bit has meaning to the originating FRTE, who is essentially being asked to slowdown transmission of the data. The BECN bit's meaning, to the destination FRTE, is a request to slow down its frame transmission. But, because of the vague standard formulation, in each specific case of Frame-Relay implementation, the following questions have to be asked:

- How does the selected equipment deal with exceeding of the CIR?
- Are the FECN/BECN/DE bits set?
- Does the FRTE honor the FECN/BECN/DE bits?

Although, most of Frame-Relay carriers provide FECN, BECN, and DE indications, some Frame-Relay switches do not include the FECN/BECN warnings to attached FRTEs, so that they can react to congestion conditions: they may just wait until the "committed information rate" is exceeded and then discard everything. DE in a Frame-Relay network comes into play when a network is severely congested. Dropping the discard-eligible frames will relieve congestion in the network so that frames not eligible for discard have a better chance of passing successfully through the network. It is the responsibility of the FRTE to set the DE bits on. The FRTE sets the DE bit so that the *Frame-Relay Frame Handler* (FRFH) knows which frames to discard in order to protect itself in case of severe congestion; however, if the FRFH's DLCI queue is full and yet the FRTE continues sending frames with no DE set, then the FRFH will get into the act and mark all frames from that FRTE as DE, notifying the FRTE that it has exceeded its committed burst.

To be more specific: during a "Mild Congestion" most SNA users of a Frame-Relay WAN will slow down (by reducing the "window" size) in order to avoid frame loss and the need to re-transmit. Most of the known Frame-Relay TCP/IP users ignore the "Mild Congestion" and continue to send frames until "Severe Congestion" occurs on their DLCI. Then, during "Severe Congestion" FRFH discards TCP/IP frames, so that frame loss will cause TCP/IP to slow down.

So "technically" speaking, response time for the "polite" SNA Frame-Relay users will suffer during mild congestion; however, they are less likely to get into severe congestion trouble. However, "impolite" TCP/IP users that cannot throttle-down during mild congestion will most definitely get into severe congestion trouble causing frame losses, and consequently, re-transmissions. The bottom line is, capacity planning in a consolidated multi-protocol Frame-Relay network becomes truly challenging.

SNA/APPN as the Consolidated Network Backbone

The vast majority of SNA shops are still reluctant to convert their SNA backbones to TCP/IP, protecting performance of mission-critical applications. Both DLSw and RFC 1490 are some of the possible avenues for network integration, but as mentioned before, neither one can resolve all the shortcomings of the less dependable protocols and fully assure reliability and performance for SNA based mission-critical applications.

This prompted some vendors to take a different approach for network consolidation. Their solutions provide SNA based gateways and "routers" to allow utilization of SNA backbones for non-SNA traffic. This approach permits capitalization on investments that many corporations have already made in their SNA backbones. It also grants the users a better utilized and more cost-effective SNA network, supporting many new applications. For example, end-users on IPX LANs can now access and communicate with other IPX LANs worldwide across SNA networks. Any client–server application, that runs on a Novell LAN, can now run over an SNA network. The new products will not affect the SNA backbone, protecting it by filtering out broadcasts from LAN based protocols.

Users on NetBIOS LANs can now also access and communicate with other NetBIOS LANs worldwide across SNA networks. Once again, any NetBIOS application (i.e., Lotus Notes, cc:Mail, MsMail, etc.) will safely run over an SNA network that is protected by filtering NetBIOS broadcasts. The same applies to TCP/IP applications' connectivity across SNA networks. End-users on TCP/IP networks can now communicate with other TCP/IP networks worldwide across SNA networks. And, obviously, any of the important TCP/IP applications (i.e., FTP, Telnet, etc.) will safely run over an SNA network.

Major vendors in networking consolidation

- *DLSw:* most router vendors (Cisco, Bay Networks, 3Com, IBM, Proteon, CrossCom, and many others) support DLSw RFC 1434 today.
- *RFC 1490:* in addition to the major router vendors, software based solutions provide access to Frame-Relay networks by utilizing RFC 1490.
- *LAN over SNA:* some of the notable examples for this approach are coming from major inter-networking vendors, such as Novell and IBM. A new solution to multi-protocol routing on top of SNA, came from ATLan, that recently introduced a family of *Enterprise Connectivity Node* (ECN) products.

Novell's products are designed to run on the existing NetWare, eliminating the need for a dedicated internetworking device in the branch. For example, NetWare for SAA provides access to 3270, 5250, LU6.2, and LU0 applications on mainframes and AS/400 minicomputers. It supports SDLC and X.25/QLLC.

continued on next page

Making Migration to TCP/IP Seamless *(continued)*

A different product—*NetWare SNA Links,* provides LAN-to-LAN connectivity over an SNA network, which also allows managers to use management tools, such as Novell's NetWare Management System (NMS), and administrative tools, such as SYSCON, over the SNA network. NetWare SNA Links can also be used for store-and-forward electronic mail traffic. The NetWare Multi-Protocol Router (MPR) Plus is a software-based router for PCs. It routes IPX, IP, AppleTalk, and OSI over a variety of LANs and provides source-route bridging for Token-Ring LANs. For WAN connectivity, it supports leased lines using Point-to-Point Protocol (PPP), Frame-Relay, ISDN, Switched Multimegabit Data Service (SMDS), and X.25.

Another software based solution for routing LAN based traffic over SNA is based on IBM's Networking Blueprint MPTN (Multi-Protocol Transport Network) concept. It is the so called family of AnyNet products. The most recent representative of this technology is the new 2217 Nways Multi-Protocol Concentrator (MpC). With AnyNet, non-SNA protocols (i.e., TCP/IP, IPX, etc.) are converted and then routed across the SNA/APPN WAN. At the destination point, another AnyNet node must be used to convert the messages back into their native protocol. AnyNet technology allows transparent multiprotocol LAN-to-LAN communications without requiring any modifications to the applications. In addition to the 2217 MpC product AnyNet Sockets over SNA come on different platforms: i.e., mainframes, RS/6000 AS/400 and PS/2. These products will provide multiprotocol support for TCP/IP, IPX, NetBIOS (and, of course SNA), over SNA/APPN WAN backbones along with LAN network management support over Frame-Relay, X.25, or SDLC links.

But a new and interesting approach of concurrent utilization for both SNA and non-SNA backbones is implemented in the ECN. ECN is based on a Concurrent Backbone Architecture (CBA) technology, that addresses parallel utilization of SNA and non-SNA backbones to transfer LAN based traffic. It is capable of concurrent utilization of both SNA and non-SNA infrastructures for data transfer. ECN supports connectivity to all popular LAN and WAN protocols, including IP, IPX/SPX, NetBIOS, AppleTalk, DECnet, Token-Ring, Ethernet, SDLC, PPP, X.25, Frame-Relay, ATM, ISDN, and other. In fact, the optimal route is selected in real time, based on the current status for the SNA and non-SNA links. The optimal route will be dynamically selected on the basis of Quality of Service (QoS) and additional criteria, with full prioritization between SNA and LAN based traffic for any given session. The optimal path is chosen among all potential end-to-end routes, available at this time, regardless of whether the routes are SNA or non-SNA based. And as it's customary for SNA implementations, communication lines can be effectively utilized above 90%.

The ECN solution is especially attractive for users currently running a traditional SNA wide area network backbone and users planning migration to TCP/IP backbone infrastructure. Although many of them already have a multiprotocol network in future years, it is expected that all of them will still have SNA/APPN as the predominant network architecture. By providing a multiprotocol solution based on the advantages of SNA/APPN, the requirements for transporting multiple protocols across a WAN are satisfied. Many network managers are reluctant to discard their SNA/APPN experience or trust transport of mission-critical data over IP-based router solutions.

Summary

Creating redundant networking infrastructure for deterministic, bandwidth-efficient mission-critical SNA applications and for non-deterministic, bandwidth-"hungry" LAN and IP networks with excess capacity for non-SNA links is probably one of the most conventional approaches by network managers. Because of their worries about congestion—the main cause of data loss and unpredictable performance—the main motive for most network managers today is: "Better safe than sorry!" However, with solutions, that efficiently utilize the SNA/ APPN architecture, SNA-centric networks will not succumb to congestion. Therefore, the costly solution of ordering excess capacity for the links can be avoided. Such solutionss are especially attractive for users, currently running a traditional SNA or SNA/APPN wide area network backbone. Although many already have a multiprotocol network, it is expected that all of them will still have SNA/APPN as the predominant network architecture for mission-critical applications in the foreseeable future. By providing a multiprotocol solution, based on the advantages of SNA/APPN, the requirements for transporting multiple protocols across a WAN can be satisfied. Many network managers are reluctant to neglect their SNA/APPN experience and trust transport of mission-critical data over IP-based router solutions. Managers are also looking to develop a seamless route for transition to TCP/IP in the future.

References

[1] R. Dixon, D. Kushi, "Data Link Switching: Switch-to-Switch Protocol," RFC 1434, March 1993.

[2] L. Wells, A. Bartky, "Data Link Switching: Switch-to-Switch Protocol AIW DLSw RIG: DLSw Closed Pages, DLSw Standard Version 1.0," RFC 1795, April 1995.

[3] T. Bradley, C. Brown, A. Malis, "Multiprotocol Interconnect over Frame Relay," RFC 1490, July 1993.

[4] E. Rabinovitch, "LAN Over SNA—Making It Easier," *Client/Server Computing,* August 1995.

[5] Clark, Wayne, "SNA Internetworking," *ConneXions,* Volume 6, No. 3, March 1992.

[6] Joyce, Steven T. and Walker II, John Q., "Advanced Peer-to-Peer Networking (APPN): An Overview," *ConneXions,* Volume 6, No. 10, October 1992.

[7] Clark, Wayne, "Accommodating SNA Peer-to-Peer Networking in a Multiprotocol Environment," *ConneXions,* Volume 7, No. 3, March 1993.

[8] Tittel, E., "Back to Basics: SNA," *ConneXions,* Volume 10, No. 4, April 1996.

[9] Moy, J., "OSPF Version 2," RFC 1583, March 1994.

[10] Hedrick, C., "Routing Information Protocol," RFC 1058, June '88.

EDDIE RABINOVITCH is a Senior Manager with the Worldwide Network and Desktop Consulting practice, Global Customer Services at Unisys Corp. He has twenty years of experience in Information Technology and Data Processing. Prior to joining Unisys, Eddie held senior technical, consulting, and managerial positions with major corporations, including IBM, Dreyfus Corp., Dun and Bradstreet, and Chemical Bank. He has published more than 50 papers with several trade and technical publications. Eddie is also a member of the editorial review board for several magazines and The Computer Measurement Group (CMG) and a frequent speaker at national and international conferences. He can be reached via e-mail to: `us000318@pop3.interramp.com`

Book Reviews

This is a collective review of four new Java books in a series from SunSoft and Prentice Hall:

- *Core Java,* ISBN 0-13-565755-5, by Gary Cornell and Cay S. Horstmann.
- *Java by Example,* ISBN 0-13-565763-6, by Jerry R Jackson and Alan L. McClellan.
- *Just Java,* ISBN 0-13-565839-X, by Peter van der Linden.
- *Instant Java,* ISBN 0-13-565821-7, by John A. Pew.

Let's start with the last first, and progress backwards.

Instant Java

Instant Java is essentially a programmer/creator/author's guide to using Java to create active Web pages. It is a 340 page guided tour through writing and running "applets" (the name given to Java programs that are interpreted in the context of a virtual machine run inside a Web browser, rather than as standalone compiled or interpreted programs running on the machine the author wrote them on). [See also `http://www.vivids.com`].

The book covers audio, graphics and animation, and provides a slew (or raft, or tranche or whatever your favourite non-specific collective noun is) of example programs.

Chapter 1 is a brief intro to Java; chapter 2 is on fundamental applets, chapter 3 on text, chapter 4 on Images, chapter 5 on animation and chapter 6 on miscellaneous. The book is very useful, but not really more than a reference work, and not one for the beginner Java programmer, so let's move on to one that is (and a nice companion work).

Just Java

Just Java is an introduction to the programming language, including a history of Object Oriented Programing, explaining some of the design tradeoffs that have gone into decisions about Java's form, in the areas of overloading, polymorphism and multiple inheritance (or lack thereof, based on bad experiences in previous languages!). This is very much a "sit-down-and-read" cover-to-cover book, and it is written in a nice friendly, informal fashion, (first person experience/sentences abound).

Chapter 1 is an intro to the Web and Java, and the synergy between the two. Chapter 2, wittily titled the "Story of O," is the aforementioned history of Object Oriented Programming. Chapter 3 introduces the base Java language; chapter 4 covers larger structures (packages, classes, visitibility, interfaces etc); chapter 5 looks at use of Arrays and Threads, Memory Management (the Garbage Collector); chapter 6 has some notes on miscellaneous problems such as security management and linking Java programs with "native" (e.g., **C** or **C++**) code fragments; chapter 7 looks at Java libraries such as I/O and networking and chapter 8 covers the *Abstract Window Toolkit* (AWT, a rather cutdown version of something akin to *Tcl*'s *Tk*). Chapter 9 looks at some future directions for Java, especially of interest to this reviewer are the areas of distributed Java programs and security and hardware implementations.

Java by Example

Java by Example, the third book that I looked at, would make an excellent sequel to *Just Java* (possibly read before *Instant Java,* or after—depends on how fast you want to start programming, or putting up active Web pages, really).

This is written in a very readable way again, but a little less informally than *Just Java*. It goes into a lot more detail about the language and use and each chapter is peppered (or salted, or seasoned) with excellent examples and code fragments. For example chapter 4 on Memory and Constructors uses sample programs with linked lists, while chapter 5 on Interfaces shows the way to structure a program from a number of classes such as a Tree and Treenode, to build up a Tree Sorter. Further chapters cover Arrays, Exception Handling, I/O, runtime types, detailed examples of native **C** code calling, threads and so forth. A second part to the book covers applets and event/thread handling in applets, forms (as per form entry—a very common requirement on Web pages now—an example includes talking to an Oracle database) and animation. Finally, appendices cover OO Programming elements of style, and a quick reference guide to the language.

Core Java

This leads us nicely to the fourth book, *Core Java*. This is altogether a larger effort, comprising some 622 pages, quite encyclopedically documenting the Java language. This is probably going to be a baseline text for Java programmers. It includes several nice publisher's tricks to increase the usefulness of the book. For example, throughout the book, a set of icons appear to indicate "aside notes," API questions, tips, danger points, hints for programmers specifically expert in Visual Basic, or **C++** previously. I found these tremendously helpful (as a **VC++** programmer!).

The book covers the same territory as *Java by Example,* in a similar order, but in much more detail. Chapter 1 is an outline of the language rationale; chapter 2 looks at the programming environment (edit/compile/run/debug cycle). Chapter 4 looks at basic Java programs, data types, statements, relations, and control flow. Chapter 4 looks at Objects and Classes; chapter 5 covers Java inheritance. Chapter 6 and 7 detail use of AWT, and design of user interfaces using AWT (I personally found this handy, as I've never done a course on user interface design!). Chapter 8 covers Applets, including access to URLs, multimedia and so on. Chapter 9 is a sequence of examples of implementations of common useful data structures such as vectors, bitsets, hash tables, linked lists, stacks, multi-dimension arrays, etc. Chapter 10 is all about exceptions and debugging (jdb). Chapters 11 and 12 cover the I/O and threads packages and chapter 13 covers networking (including writing servers and CGI scripts and some concerns about security).

An appendix covers installing the software from the CD-ROM associated with this (and the other) book. All the books are adequately indexed (my usual set of Java questions were answered easily enough via the index or the table of contents).

My recommendation

These books form a very important reference set. I still find the SAMS *Teach Yourself Java in 21 Days* the best startup book, especially for students, and the O'Reilly *Java in a Nutshell* book is a good software engineer's reference work too (and very aggressively priced!). But these books are from the "horse's mouth" as it were, and with the CD, make a nice set. So they'll be on my bookshelf without a doubt.

The CD

All four books come equipped with a CD (the same CD), with the *Java Developer's Kit* (JDK) 1.0, for Win95, WinNT and Solaris, Café Lite, WinEdit customized for Java (if you aren't a UNIX/Emacs user :–) and WinZip and a beta JDK for a Macintosh, as well as example applets and Web pages. See also `http://www.prenhall.com/~java_sun`

—*Jon Crowcroft, University College London*
`J.Crowcroft@cs.ucl.ac.uk`

First Announcement and Call for Papers

JENC8, the *8th Joint European Networking Conference* will be held in Edinburgh, May 12–15, 1997. The event is organised by TERENA, the Trans-European Research and Education Networking Association, hosted by UKERNA, the United Kingdom Education and Research Networking Association, with local organisation by the University of Edinburgh, and with the local assistance from Concorde Services Ltd.

Theme

The main theme of the JENC8 conference will be "Diversity and Integration: The New European Networking Landscape." 1997 will mark an important event for the European networking community, namely the final deregulation steps of much of the European telecommunication infrastructure. In parallel, Europe is witnessing an explosion of demand for network services and innovative distributed applications. A large number of corresponding projects and pre-competitive development efforts are currently underway, in order to strengthen the European position in a global, competitive and deregulated environment. In this context, and building on the established tradition of previous JENCs, JENC8 will focus on the effects of these changes and developments for researchers, Network and Service Providers and commercial users, considering technical, economic and political issues, as well as user support, training and educational matters. JENC8 will be *the* European forum to get up-to-date information, to debate and assess the new deregulated telecommunication environment, leading-edge applications, and the network/internetwork support infrastructure which is currently being developed.

Topics

• *Emerging Network Technologies and Network Engineering:* New network technologies and engineering methods are currently applied in order to satisfy the growing demand for bandwidth and service quality. Contributions are invited discussing deployment experiences with high-speed networks & protocols, routing issues, IPv6 migration, multicasting, resource reservation protocols, and mobility issues.

• *User Support, Training & Education:* The on-going shift from mostly computer science- and network-related user groups towards broader terms of network usage and new customer communities requires substantial efforts in terms of support, training and continued education. Contributions are invited for, but are not limited to, virtual education and learning communities, 2nd level education, support and training tools and experiences, as well as library/content-provider integration.

• *Security and Management Issues:* As more and more business applications are utilised over open networks, the provision of reliable, scalable, secure and manageable services is becoming a crucial success factor. Especially "hot topics" for JENC8 are management and security aspects of the interconnection of inter- and intranets, secure applications (e.g., payment or purchase systems), firewall technologies, new security services and protocols, as well as management issues in inhomogeneous network environments.

• *Information Systems and Distributed Applications:* With the broad availability and growing interoperability of networked information systems, more and more complex distributed applications emerge, that are suited to let users provide and locate/consume information, and to foster interpersonal and group-based cooperation. Contributions of interest include new ways of information provision, location, indexing, and evaluation on the World-Wide Web, interconnection of internet/intranet information bases, programming environments and tools for distributed applications, as well as products and services to support distributed cooperative work.

• *Economic and Political Issues:* A variety of non-technical issues have emerged with respect to the changed usage of networks and distributed applications, such as funding and revenue-sharing models for network services, usage of existing networks such as the Internet for commercial purposes, applicability of existing laws covering intellectual property rights, privacy and data protection issues, national regulation efforts concerning the use of cryptography etc. Contributions are sought from all parties concerned with such issues, including telecommunication industry, content/access providers, lawmakers, law-enforcement agencies, and user/interest groups.

Submissions

Full papers (maximum of 4,000 words) are to be submitted. All papers must be written in English. Electronic submission is highly recommended and should take place as follows:

- ASCII or uuencoded *PostScript* to: `jenc8-submit@terena.nl`
- *PostScript* documents: anonymous FTP to `erasmus.terena.nl` into directory `pub/jenc8/submit`. Please note that files placed in this directory can only be written once and cannot be deleted afterwards.

Should electronic submission be impossible, please submit 6 copies of double-spaced full paper manuscript (maximum of 4,000 words) with an abstract to the JENC8 Secretariat at the address given below.

Exhibits and demonstrations

An exhibition area will be available for international and national companies and institutions for demonstration of their products and services. Requests for exhibition space should be submitted by a description not exceeding one page. There will also be the opportunity for participants from academia and industry to present exciting applications of networking services etc. in the form of a demonstration. Proposed demonstrations should be documented with a description not exceeding one page.

Tutorials

A number of tutorials are planned to be held before or during the JENC8 conference. Proposals for tutorials should be documented by a description not exceeding one page.

Important dates

November 10, 1996:	Full manuscripts and tutorials/demonstration proposals due.
January 31, 1997:	Notification of acceptance of papers/proposals, and requests for exhibition space due.
March 16, 1997:	Camera-ready papers due.

Publication

Printed conference proceedings containing full papers will be included in the delegates pack distributed at the conference. A selection of the best presented papers will be published in an appropriate journal.

Venue

Edinburgh is one of the world's most beautiful capital cities, renowned for its unique heritage, architectural grandeur and cultural vibrancy. The venue for JENC8 is the new Edinburgh International conference Centre, which is located in the heart of the city. The city and the surrounding are have an abundance of attractions to suit everyone. A range of accommodation has been booked on behalf of the Conference Centre.

More information

JENC8 Secretariat
Singel 466–468
NL 1017 AW Amsterdam
The Netherlands
E-mail: `jenc8-sec@terena.nl`
URL: `http://www.terena.nl/terena/jenc8/`

Call for Papers

The Internet Society *Symposium on Network and Distributed System Security* will be held February 10–11, 1997, San Diego Princess Resort in San Diego, California.

Goal

The symposium will bring together people who are building hardware and software to provide network and distributed system security services. The symposium is intended for those interested in the practical aspects of network and distributed system security, focusing on actual system design and implementation, rather than theory. We hope to foster the exchange of technical information that will encourage and enable the Internet community to apply, deploy, and advance the state of available security technology. Symposium proceedings will be published by the IEEE Computer Society Press.

Topics

Topics for the symposium include, but are not limited to, the following:

- Design and implementation of communication security services: authentication, integrity, confidentiality, authorization, non-repudiation, and availability.
- Design and implementation of security mechanisms, services, and APIs to support communication security services, key management and certification infrastructures, audit, and intrusion detection.
- Requirements and designs for securing network information resources and tools—World-Wide Web (WWW), Gopher, archie, and WAIS.
- Requirements and designs for systems supporting electronic commerce—payment services, fee-for-access, EDI, notary—endorsement, licensing, bonding, and other forms of assurance.
- Design and implementation of measures for controlling network communication—firewalls, packet filters, application gateways, and user/host authentication schemes.
- Requirements and designs for telecommunications security especially for emerging technologies—very large systems like the Internet, high-speed systems like the gigabit testbeds, wireless systems, and personal communication systems.
- Special issues and problems in security architecture, such as interplay between security goals and other goals—efficiency, reliability, interoperability, resource sharing, and cost.
- Integration of security services with system and application security facilities, and application protocols—including but not limited to message handling, file transport, remote file access, directories, time synchronization, data base management, routing, voice and video multicast, network management, boot services, and mobile computing.

Submissions

The program committee invites technical papers and panel proposals for topics of technical and general interest. Technical papers should be 10–20 pages in length. Panel proposals should be two pages and should describe the topic, identify the panel chair, explain the format of the panel, and list three to four potential panelists. Technical papers will appear in the proceedings. A description of each panel will appear in the proceedings, and may at the discretion of the panel chair, include written position statements from each panelist.

Each submission must contain a separate title page with the type of submission (paper or panel), the title or topic, the names of the author(s), organizational affiliation(s), telephone and FAX numbers, postal addresses, Internet electronic mail addresses, and must list a single point of contact if more than one author. The names of authors, affiliations, and other identifying information should appear only on the separate title page.

Submissions must be received by 1 August 1996, and should be made via electronic mail in either *PostScript* or ASCII format. If the committee is unable to print a *PostScript* submission, it will be returned and hardcopy requested. Therefore, *PostScript* submissions should arrive well before 1 August. If electronic submission is difficult, submissions should be sent via postal mail. All submissions and program related correspondence (only) should be directed to the program chair:

Clifford Neuman,
University of Southern California, Information Sciences Institute
4676 Admiralty Way
Marina del Rey, California 90292–6695
Phone: +1 (310) 822-1511
FAX: +1 (310) 823-6714
E-mail: `sndss97-submissions@isi.edu`

Final call for papers, program, and registration information will be available at the URL: `http://www.isoc.org/conferences/ndss97`

Each submission will be acknowledged by e-mail. If acknowledgment is not received within seven days, please contact the program chair as indicated above.

Important dates

Submissions due:	August 1, 1996
Notification to Authors:	October 1, 1996
Camera-Ready Copy due:	November 1, 1996

Write to *ConneXions*!

We'd love to hear your comments, suggestions and questions about anything you read in *ConneXions*. Our editorial address is given below. Use it for letters to the Editor, requests for the index of back issues, questions about particular articles etc.:

ConneXions—The Interoperability Report
303 Vintage Park Drive
Foster City, California 94404–1138, USA
Phone: +1 415-578-6900
Internet: `connexions@interop.com` `http://www.interop.com`

Subscription information

For questions about your subscription please call our customer service hotline: 1-800-575-5717 or +1 610-892-1959 outside the USA. The fax number is +1 610-565-1858. The mailing address for subscription payments is: P.O. Box 976, Media, PA 19063–0976.

CONNEXIONS
303 Vintage Park Drive
Suite 201
Foster City, CA 94404-1138
Phone: 415-578-6900
FAX: 415-525-0194

ADDRESS CORRECTION
REQUESTED

FIRST CLASS MAIL
U.S. POSTAGE
PAID
SAN JOSE, CA
PERMIT NO. 1

CONNEXIONS

EDITOR and PUBLISHER — Ole J. Jacobsen